Multiobjective fuzzy clustering approach based on tissue-like membrane systems

Hong Peng [a], Peng Shi [b], Jun Wang [c,*], Agustín Riscos-Núñez [d], Mario J. Pérez-Jiménez [d]

[a] School of Computer and Software Engineering, Xihua University, Chengdu, Sichuan, 610039, China
[b] School of Electrical and Electronic Engineering, University of Adelaide, Adelaide, SA 5005, Australia
[c] School of Electrical Engineering and Electronic Information, Xihua University, Chengdu, Sichuan, 610039, China
[d] Research Group of Natural Computing, Department of Computer Science and Artificial Intelligence, University of Seville, Sevilla, 41012, Spain

Keywords:
Fuzzy clustering
Multiobjective clustering problem
Membrane computing
Tissue-like membrane system

A B S T R A C T

Fuzzy clustering problem is usually posed as an optimization problem. However, the existing research has shown that clustering technique that optimizes a single cluster validity index may not provide satisfactory results on different kinds of data sets. This paper proposes a multiobjective clustering framework for fuzzy clustering, in which a tissue-like membrane system with a special cell structure is designed to integrate a non-dominated sorting technique and a modified differential evolution mechanism. Based on the multiobjective clustering framework, a fuzzy clustering approach is realized to optimize three cluster validity indices that can capture different characteristics. The proposed approach is evaluated on six artificial and ten real-life data sets and is compared with several multiobjective and singleobjective techniques. The comparison results demonstrate the effectiveness and advantage of the proposed approach on clustering the data sets with different characteristics.

1. Introduction

Clustering as a class of machine learning techniques has been widely used in many fields, such as pattern recognition, image pro- cessing, data mining and bioinformatics [1,2]. Clustering is the task of finding natural partitioning within a data set such that patterns within the same cluster are more similar than those within differ- ent clusters. Fuzzy c-means (FCM) [3] and k-means [4] are two of the most popular clustering algorithms, in which data clustering is regarded as an optimization problem and total cluster variance J_m is used as the objective function to be optimized. However, the two classical algorithms have a notable shortcoming: they easily fall into local minima and may not converge to the global minima [5]. To overcome this shortcoming, some global optimization tech- niques have been introduced to deal with data clustering problems in the past years, for example, simulated annealing (SA)-based [6], differential evolution (DE)-based [7–9], black hole-based [10], par- ticle swarm optimization (PSO)-based [11–13], artificial bee colony (ABC)-based [14], genetic algorithms (GA)-based [15–17] and ant colony optimization (ACO)-based clustering algorithms [18] as well as a gravitational clustering algorithm [19]. In these clustering al- gorithms, a variety of cluster validity indices have been used to evaluate the goodness of partitioning obtained by them, such as Sym-index [20], *I*-index [21] and XB-index [22]. The existing works have indicated that these cluster validity indices have different characteristics, for example, XB-index is very suitable for processing the hyperspherical shaped clusters [8,9], while Sym-index that is more useful to detect symmetrical sharped clusters [23].

Most optimization-based clustering algorithms are single-objective because only a single validity measure is optimized. Note that a single validity measure can only reflect some intrinsic partitioning properties, for example, the compactness of clusters, the spatial separation between the clusters and the cluster's symmetry. Therefore, some clustering algorithms that use J_m as the objective function are only able to find compact hyperspherical and convex clusters like k-means. If clusters with different geometric shapes are present in the same data set, the clustering algorithms that use a single cluster validity index will fail to deal with the data set. Therefore, it is required to simultaneously optimize several cluster validity indices that can capture different data characteristics. Based on this consideration, data clustering should be viewed as a multiobjective optimization problem. Several works on multiobjective clustering have been published recently. Handl and Knowles [24] proposed a multiobjective clustering technique, *MOCK*, which can recognize the appropriate partitioning from the data sets that contain either hyperspherical shaped clusters or well-separated clusters. But, it fails to detect overlapping clusters

that have different shapes rather than hyperspheres. Another disadvantage of *MOCK* lies in its encoding, which can cause that the length of each chromosome will increase largely with the increase in the number of points. Faceli et al. [25] presented a clustering algorithm that combined cluster ensemble and multi-objective clustering techniques. Saha and Bandyopadhyay [26] presented a multiobjective clustering technique, called *VAMOSA*, where center-based encoding was used. Two cluster validity indices were optimized simultaneously: an Euclidean distance-based XB-index, and another point symmetry distance-based Sym-index. Experimental results indicated that *VAMOSA* can evolve the appropriate partitioning from data sets having clusters of any shape, size or convexity. The multiobjective clustering technique proposed in Saha and Bandyopadhyay [27], *GenClustMOO*, used a simulated annealing-based multiobjective clustering technique as the underlying optimization strategy. In Saha et al. [28], a multiobjective modified differential evolution-based fuzzy clustering, *MOmoDEFC*, has been addressed, in which both XB-index and FCM measure (J_m) were used as two objective functions. Simulation results showed that *MOmoDEFC* can optimize both the compactness and separation of clusters simultaneously. However, it may not be perfect to process the data set having cluster symmetry.

Membrane computing, initiated by Păun [29], is a class of distributed parallel computing models, known as membrane systems or P systems. The novel computing models were inspired from the structure and functioning of living cells as well as the cooperation of cells in tissues, organs and populations of cells [30–37]. In recent years, the inherent advantages and characteristics that membrane systems possess have attracted much attention on applications of membrane computing [38–43], for example, membrane algorithms for solving optimization problems. The research results on a variety of optimization problems have exhibited the potentiality of membrane computing models in the following three aspects: better convergence, stronger robustness and better balance between exploration and exploitation [44–49].

Based on the above consideration, the main motivation of this work is using membrane systems to develop a multiobjective optimization framework for fuzzy clustering problems. The role of the tissue-like membrane system stays in three aspects: (i) integrating differential evolution mechanism, (ii) realizing non-dominated sorting strategy, and (iii) realizing the coevolution between the objects in different cells. According to the multiobjective framework, three cluster validity indices, J_m, XB-index and Sym-index, are selected as objective functions, and a novel multiobjective fuzzy clustering approach is proposed in this paper, called *MOFC-TMS*. In Peng et al.[50], an evolution-communication membrane system has been used to propose a fuzzy cluster approach, called *Fuzzy-MC*. However, Fuzzy-MC is single-objective because only XB-index is considered as objective function to be optimized. In contrast to *Fuzzy-MC, MOFC-TMS* has two differences: (i) membrane systems are considered to solve multiobjective fuzzy clustering problems; (ii) a tissue-like membrane system with a special membrane structure is considered and a modification of differential evolution mechanism is designed according to the special structure. In addition, a single-objective approach that uses the special membrane structure and the modified differential evolution mechanism is implemented in simulation, which has better clustering performance over *Fuzzy-MC*. To the best of our knowledge, this is the first attempt to use a membrane computing model to solve multiobjective fuzzy clustering problems.

The rest of this paper is arranged as follows. Section 2 introduces multiobjective fuzzy clustering problems. Section 3 briefly reviews the definition and inherent mechanism of tissue-like membrane systems. In Section 4, a multiobjective clustering framework for fuzzy clustering is described in detail. In Section 5, experimental results carried out on some benchmark data sets are presented. Finally, conclusions are drawn in Section 6.

2. Problem statement

Data clustering in a d-dimensional Euclidean space is a process, which partitions n data points into several groups according to some similarity. Suppose that $X = \{X_1, X_2, \ldots, X_n\}$ is a data set consisting of n unlabeled data points, where $X_i = (x_{i1}, x_{i2}, \ldots, x_{id})$, $i = 1, 2, \ldots, n$. A fuzzy clustering approach tries to find a fuzzy partitioning, $\{C_1, C_2, \ldots, C_K\}$, such that the similarity of the data points in the same cluster is maximum and data points from different clusters differ as much as possible. Note that for a fuzzy partitioning, a data point can belong to all classes with a certain fuzzy membership degree for each class. Therefore, an appropriate partitioning matrix, $U = [u_{ij}]_{K\times n}$, needs to be evolved, where $u_{ij} \in [0, 1]$ denotes the membership grade of the jth element to the ith cluster. The fuzzy partitioning should maintain the following properties:

$$\begin{cases} 0 < \sum_{j=1}^{n} u_{ij} < n & \text{for } i = 1, 2, \ldots, K \\ \sum_{i=1}^{K} u_{ij} = 1 & \text{for } j = 1, 2, \ldots, n \\ \sum_{i=1}^{K}\sum_{j=1}^{n} u_{ij} = n \end{cases} \tag{1}$$

In the existing optimization-based clustering algorithms, fuzzy clustering problem is regarded as an optimization problem, however, most of them are single-objective because only a single cluster valid index is optimized. Note that a cluster valid index focuses mainly on some intrinsic partitioning property. However, a data set may have different geometric shapes, for example, the compactness of clusters, the spatial separation between the clusters and the cluster's symmetry. So, a single cluster valid index can fail to deal with data sets that have different geometric shapes. Therefore, fuzzy clustering problem should be posed as a multiobjective optimization problem, in which more objective functions (cluster valid indices) are optimized simultaneously. There are three cluster valid indices used widely in single-objective clustering algorithms: J_m, XB-index and Sym-index. The existing results have shown that the three indices can capture different data characteristics: (i) J_m can detect hyperspherical shaped clusters; (ii) XB-index can well detect compact and hyperspherical shaped clusters and emphasize the separation between two nearest clusters; (iii) Sym-index is more effective to detect symmetrical sharped clusters from the data set.

Since the three cluster valid indices can better capture the intrinsic characteristics of samples, they will be used as the objective functions to be optimized simultaneously in this work. Thus, fuzzy clustering problem can be formally defined as a multiobjective minimization problem

$$\begin{cases} \min\limits_{(z_1,\ldots,z_K)} [f_1, f_2, f_3] \\ f_1(z_1, \ldots, z_K) = J_m(z_1, \ldots, z_K) \\ f_2(z_1, \ldots, z_K) = XB(z_1, \ldots, z_K) \\ f_3(z_1, \ldots, z_K) = 1/Sym(z_1, \ldots, z_K) \end{cases} \tag{2}$$

where $z_1, \ldots, z_K$ are K parameters to be optimized, which denote K cluster centers of a partitioning.

In FCM, J_m is defined as follows

$$J_m(Z) = \sum_{i=1}^{K}\sum_{j=1}^{n} u_{i,j}^2 d^2(x_j, z_i) \tag{3}$$

where $Z = (z_1, \ldots, z_K)$, $d(x_j, z_i)$ is the distance between x_j and z_i, and $u_{i,j}$ is computed by

$$u_{i,j} = \left(\frac{1}{d(x_j, x_i)}\right) \Big/ \left(\sum_{i=1}^{K} \frac{1}{d(x_j, x_i)}\right)$$

Generally, a lower value of J_m implies a better clustering solution.

XB-index is defined as a function of the ratio of the total variation σ to the minimum separation *sep* of the clusters [22],

$$XB(Z) = \frac{\sigma(Z;X)}{n \cdot sep(Z)} = \frac{\sum_{i=1}^{K}\sum_{j=1}^{n} u_{i,j}^2 d^2(x_j, z_i)}{n \cdot \min_{i \neq j} ||z_i - z_j||^2} \tag{4}$$

where $||\cdot||$ is the Euclidean norm. Note that when the partitioning is compact and good, σ should be low while *sep* should be high, thereby yielding lower values of the *XB* index.

Sym-index is a point symmetry distance-based cluster valid index [20,23]. It is defined as follows:

$$Sym(Z) = (\frac{1}{K} \times \frac{1}{\mathcal{E}_K} \times D_K) \tag{5}$$

where K is the number of clusters, and

$$\mathcal{E}_K = \sum_{i=1}^{K} E_i = \sum_{i=1}^{K}\sum_{j=1}^{n} u_{i,j} d_{ps}(x_j, z_i)$$

$$D_K = \max_{i,j=1}^{K} ||z_i - z_j||$$

where $d_{ps}(x_j, z_i)$ is the point symmetry distance between x_j and z_i. Here, the first *knear* nearest neighbors of $x_j^* = 2 \times z_i - x_j$ will be searched among only those points which are in cluster i, i.e., the *knear* nearest neighbors of x_j^*, the reflected point of x_j with respect to z_i, and x_j should belong to the ith cluster. Note that for clusters which have good symmetrical structure, $\mathcal{E}_K$ value is low. D_K measures the maximum separation between a pair of clusters, which is bounded by the maximum separation between a pair of points in the data set. Therefore, *Sym* needs to be maximized for optimal clustering.

3. Tissue-like membrane systems

In this section we briefly review the definition and inherent mechanism of tissue-like membrane systems. A more detailed description of tissue-like membrane systems can be found in [32,51].

A tissue-like membrane system (of degree $q > 0$) with symport/antiport rules is formally defined as a tuple

$$\Pi = (O, w_1, \ldots, w_q, R_1, \ldots, R_q, R', i_0)$$

where

(1) O is a finite alphabet, whose symbols are called objects;
(2) $w_1, \ldots, w_q$ are initial multisets of objects;
(3) R_i are finite sets of evolution rules in cell i, $1 \leq i \leq q$;
(4) R' is a finite set of communication rules of the form $(i, u/v, j)$, $i \neq j$, $i, j = 0, 1, 2, \ldots, q$, $u, v \in O^*$;
(5) $i_0 \in \{0, 1, 2, \ldots, q\}$ indicates the output region of the system.

The tissue-like membrane system consists of q cells, and each cell is surrounded by a cell membrane. The outer region of the q cells is called the environment. Usually, each cell contains one or more objects. $w_1, w_2, \ldots, w_q$ denote the multisets of objects of the q cells, respectively. It is assumed here that in the environment any object is available.

There are two types of rules: evolution rules and communication rules. Each cell usually contains one or more evolution rules, while communication rules are built between two different cells or between a cell and the environment. An evolution rule is of the form $u \to v$, application of which means that multiset of objects u is evolved to multiset of objects v. Communication rule of the form $(i, u/v, j)$ is called as an antiport rule, and application of such rule means that multiset of objects u in cell i and multiset of objects v in cell j are interchanged. Note that if either $i = 0$ or $j = 0$ then the two multisets of objects are interchanged between a cell and the environment. If one of u or v is the empty multiset λ, the communication rule is called as a symport rule, for example, $(i, u/\lambda, j)$, which means that multiset of objects u will be communicated from cell i to cell j. By these communication rules, the tissue-like membrane system can be described as a virtual graph where the q cells denote the nodes and the edges indicate if it is possible for pairs of cells to communicate directly.

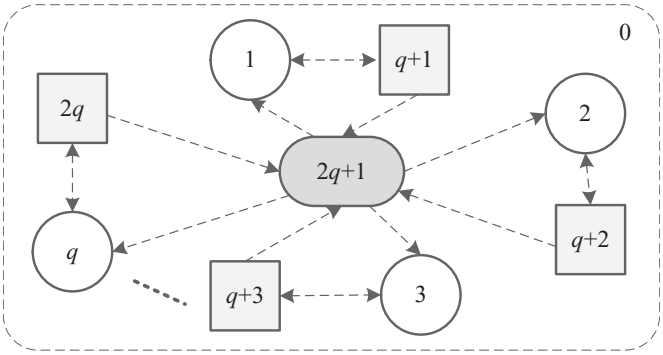

Fig. 1. The designed tissue-like membrane system and communication channels of objects between cells, where circles, squares and the rounded rectangle denote evolution cells, local memory cells and global memory cell, respectively.

As usual in membrane computing, the q cells as computing units work in parallel (a universal clock is considered here). The tissue-like membrane system starts with initial multisets $w_1, w_2, \ldots, w_q$. And then, in each step, something happens: the objects in cells are evolved and some of them are communicated. The process is repeated until the halting condition is satisfied. When it halts, the system produces a final result in the output region.

4. Multiobjective clustering framework for fuzzy clustering

The main purpose of this work is to develop a multiobjective clustering framework for fuzzy clustering problem by using membrane systems. To solve fuzzy clustering problem (2), a clustering algorithm based on the multiobjective clustering framework will be realized, called *MOFC-TMS*, in which J_m, XB-index and Sym-index are optimized simultaneously. Note that the presented multiobjective framework is flexible since it can also be used to optimize the combination of other cluster validity indices.

We design a tissue-like membrane system with a special membrane structure, shown in Fig. 1. The membrane system has $(2q + 1)$ cells, and they are classified as two categories according to the different roles: evolution cells and memory cells. The cells labeled by $1, 2, \ldots, q$ are evolution cells. The role of evolution cells is to evolve the objects in the system. The cells labeled by $q + 1, q + 2, \ldots, 2q, 2q + 1$ are memory cells. The memory cells can be further divided as two types: the q cells labeled by $q + 1, q + 2, \ldots, 2q$, called local memory cells, and the cell labeled by $2q + 1$, called the global memory cell. Each evolution cell corresponds to a local memory cell. The memory cells are designed to store Pareto non-dominant objects of the corresponding evolution cells, while the global memory cell is used to store non-dominant objects of all evolution cells. Note that the concept of non-dominance is provided below.

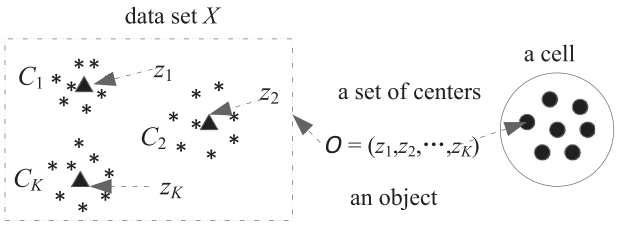

Fig. 2. The representation of objects in cells.

4.1. Objects

To solve a fuzzy clustering problem, the tissue-like membrane system is considered to optimize a set of cluster centers for the given data set. Therefore, each object in the system should be designed to express a feasible solution (a set of cluster centers). Suppose that $X = \{X_1, X_2, \ldots, X_n\} \subseteq R^{n\times d}$ is a data set to be clustered, that will be divided into K clusters, $C_1, C_2, \ldots, C_K$. Denote by $\{z_1, z_2, \ldots, z_K\}$ a set of cluster centers of the K clusters, where $z_i = (z_{i1}, z_{i2}, \ldots, z_{id}) \in R^d, i = 1, 2, \ldots, K$. Fig. 2 shows a correspondence relationship between a solution (or a set of cluster centers) of the clustering problem and an object in the cells. Thus, each object in the cells is considered as a $(K \times d)$-dimensional vector to denote a set of cluster centers:

$$O = (z_{11}, z_{12}, \ldots, z_{1d}, \ldots, z_{i1}, z_{i2}, \ldots, z_{id}, \ldots, z_{K1}, z_{K2}, \ldots, z_{Kd})$$

where $(z_{i1}, \ldots, z_{id})$ corresponds to ith cluster center z_i, $i = 1, 2, \ldots, K$. Suppose that evolution cells have the same number of objects, denoted by m. Furthermore, the global memory cell is assigned as the output region of the system. When the system halts, its objects are considered as the outputs of the entire system.

To deal with the multi-objective clustering problem, the concept of non-dominance is introduced into the tissue-like membrane system. The non-dominance of objects are defined as follows: an object O_1 is said to dominate another object O_2 if both conditions specified below are satisfied:

(i) Object O_1 is no worse than object O_2 in all objectives.
(ii) Object O_1 is strictly better than object O_2 in at least one objective.

If either of the above conditions is violated, object O_1 does not dominate object O_2.

4.2. Initialization

Initially, the tissue-like membrane system generates m initial objects for each evolution cell. When an initial object is generated, $K \times d$ random real numbers are produced to form the object with the following constraints:

$$A_1 \le z_{i1} \le B_1, \ldots, A_j \le z_{ij} \le B_j, \ldots, A_d \le z_{id} \le B_d$$

where A_j and B_j are lower and upper bounds for the jth component of data points to be clustered, $1 \le j \le d$.

Once each evolution cell completes the initialization, all its objects are calculated to fill the corresponding local memory cell according to the non-dominated sorting strategy described below. After all evolution cells are initialized, the non-dominated sorting strategy is used again to fill the global memory cell.

4.3. Communication mechanism

As usual, the tissue-like membrane system uses communication rules to exchange the objects between two different cells (evolution cells, local memory cells and global memory cell). There are two types of communication rules:

(i) Antiport rules: $(i, O_1O_2 \ldots O_s / {}_1O'_2 \ldots O'_s, q+i),\ i = 1, 2, \ldots, q$.
The rule establishes a bidirectional communication channel between an evolution cell and the corresponding local memory cell.
(ii) Symport rules: $(i, O_1O_2 \ldots O_s/\lambda, j),\ j = 2q+1$ for $i = q+1, q+2, \ldots, 2q$ or $i = 2q+1$ for $j = 1, 2, \ldots, q$.
This type rule establishes a unidirectional communication channel between an evolution cell and the global memory cell or between a local memory cell and the global memory cell.

4.4. Evolution mechanism

During computation, the objects in the system are evolved by evolution rules in the q evolution cells. In this work, mutation and crossover operations of differential evolution (DE) algorithms [52] are introduced as evolution rules of objects. In order to increase the ability to exploit, a modified mutation operation is developed according to the special structure of the tissue-like membrane system, which can be viewed as a variant of the rule "DE/current-to-best/1" in the DE [53,54]. Moreover, it can also enhance the cooperation between the objects in different cells. The modified mutation operation can be described as follows:

$$Y_i^j = O_i^j + F \cdot (O_{lbest}^j - O_i^j) + F \cdot (O_{gbest}^j - O_i^j) + F \cdot (O_{r_1}^j - O_{r_2}^j) \quad (6)$$

where $j = 1, 2, \ldots, Kd$; $O_i = (O_i^1, \ldots, O_i^{Kd})$ is an original object in evolution cell i, and $Y_i = (Y_i^1, \ldots, Y_i^{Kd})$ is the created donor object; $O_{r_1} = (O_{r_1}^1, \ldots, O_{r_1}^{Kd})$ and $O_{r_2} = (O_{r_2}^1, \ldots, O_{r_2}^{Kd})$ are two objects chosen randomly from evolution cell i; $O_{lbest} = (O_{lbest}^1, \ldots, O_{lbest}^{Kd})$ is a best object randomly selected from first level front of non-dominant objects that are communicated from the corresponding local memory cell, and $O_{gbest} = (O_{gbest}^1, \ldots, O_{gbest}^{Kd})$ is a best object randomly selected from first level front of non-dominant objects that are communicated from global memory cell; F is a scaling factor and is given by $F = 0.5 \times (1 + \mathrm{rand}(0, 1))$.

After the mutation operation, crossover operation is applied to each pair of current object and the corresponding mutant object to generate a trial object $Z_i = (Z_i^1, \ldots, Z_i^{Kd})$. The crossover operation is defined as follows:

$$Z_i^j = \begin{cases} Y_i^j, & \text{if } \mathrm{rand}_i \le C_r \text{ or } j = \mathrm{rand}_j \\ O_i^j, & \text{otherwise} \end{cases} \quad (7)$$

where $j = 1, 2, \ldots, Kd$; the crossover rate C_r is a user-specified constant within the range [0, 1], which controls the fraction of parameter values copied from the mutant object; rand_j is a randomly chosen integer in the range [1, Kd], which ensures that Z_i gets at least one component from Y_i.

4.5. Update of local memory cells

Local memory cells are designed to store non-dominant objects of the corresponding evolution cells. During evolution, m original objects in each evolution cell, $O_1, O_2, \ldots, O_m$, generate m trial objects, $Z_1, Z_2, \ldots, Z_m$. Thus, each evolution cell can have $2m$ objects after the objects are evolved. Then, m objects of them are selected from the evolution cell according to the non-dominated sorting strategy given below and are communicated into the corresponding local memory cell to update the existing non-dominated objects, whereas the remaining objects of them will be discarded. The non-dominated sorting strategy used here is originated from the known NSGA-II algorithm [55]:

(i) Non-dominated sorting. Every object is assigned a rank level according to the non-dominated concept, 1 is the best, 2 is the second, and so on. The objects with the same level belong to the same Pareto front, named $F_1, F_2, \ldots$.

(ii) Communication and truncation. Objects in F_1 are first communicated into the corresponding local memory cell, objects in F_2 are then communicated, and so on. Suppose objects in F_l are last communicated. To accurately choose m objects, the crowding distance [55] is employed to sort the objects in F_l and the best objects selected from them are communicated into the corresponding local memory cell.

4.6. Update of global memory cell

In the tissue-like membrane system, the global memory cell is considered to store non-dominant objects of the entire system. After all local memory cells have been updated, all objects in them will be chosen to update the global memory cell according to the non-dominated sorting strategy described above in next computing step. Note that the delay of a computing step exists between the objects in the global memory cell and the objects in local memory cells: non-dominant objects stored in the global memory cell are in fact global non-dominant objects in the previous computing step. The mutation rules described above indicate that a randomly selected best object in the first level front, which is communicated from the global memory cell, will participate the evolution of objects in the evolution cells. This consideration can bring the following benefits: improving the diversity of objects in the system and avoiding premature convergence.

4.7. Halting and output

The evolution-communication and update processes described above are repeated constantly until a prescribed maximum number of computation steps (or number of iterations), t_{max}, is reached. Thus, the tissue-like membrane system halts. When the system halts, all or a part of non-dominant objects on the final Pareto optimal front in the global memory cell are regarded as final solutions of the tissue-like membrane system.

Each of the final solutions provides a way of clustering the given data set, however, sometimes the user may want to obtain a single solution. In this work, a semi-supervised approach is used to select a single solution from the final non-dominant solutions, which is similar to the method used in [26]. The class level of 10% of a data set is assumed to be the known test patterns, and the remaining 90% of the data set is used to execute the proposed *MOFC-TMS*, and then a single solution is chosen according to *Minkowski score*.

Let T be the "true"solution and S be the solution that we wish to measure. Denote by n_{11} the number of pairs of elements that are in the same cluster in both T and S. Denote by n_{01} the number of pairs that are in the same cluster only in S, and by n_{10} the number of pairs that are in the same cluster only in T. *Minkowski score* is then defined as

$$MS(T,S) = \sqrt{\frac{n_{01}+n_{10}}{n_{11}+n_{10}}} \tag{8}$$

Usually, the optimum score is 0, with lower scores being better. Therefore, the solution with the minimum *Minkowski score* value calculated over the test set is selected as the best solution.

5. Experimental results and comparison study

In order to check the effectiveness of *MOFC-TMS*, the experiments on six artificial and ten real-life data sets have been conducted to compare the *MOFC-TMS* with three multi-objective clustering approaches, its single-objective versions and other single-objective clustering approaches. The data sets with different characteristics of shape, size, compactness and symmetry are described as follows.

(a) Artificial data sets:
   (1) *Sym_3_2*: This data set contains 600 data points distributed on three clusters, shown in Fig. 3(a).
   (2) *Ring_3_2*: This data set is a combination of ringshaped spherically compact and linear clusters, shown in Fig. 3(b).
   (3) *Sph_5_2*: This data set consists of 250 two dimensional data points distributed over 5 spherically shaped clusters. The clusters present in this data set are highly overlapping, each consisting of 50 data points. This data set is shown in Fig. 3(c).
   (4) *Sph_9_2*: This data set consists of 900 data points in the two-dimensional space distributed over 9 clusters. Each cluster contains 100 data points and the clusters are highly overlapping to each other. This data set is shown in Fig. 3(d).
   (5) *Sizes_5*: This data set consists of 1000 data points distributed over 4 squares, shown in Fig. 3(e).
   (6) *Square4*: This data set consists of 1000 data points distributed over 4 squared clusters, shown in Fig. 3(e).

   These data sets contain symmetrical shaped clusters (e.g. *Sym_3_2* and *Ring_3_2*: ring-shaped clusters and ellipsoidal), hyperspherical shaped clusters and highly overlapping clusters (e.g., *Sph_5_2* and *Sph_9_2*), or well-separated clusters of different shapes and sizes (e.g., *Sizes_5* and *Square4*).

(b) Real-life data sets: ten real-life data sets are retrieved from the database of University of California Irvine (UCI)[56] for machine learning; these data sets are often used to test the performance of all kinds of algorithms.
   (1) *Iris*: This data set consists of 150 points distributed over three clusters, namely, Setosa, Versicolor and Virginica. The data set is in four-dimensional space (sepal length, sepal width, petal length and petal width). Two classes (Versicolor and Virginica) have a large amount of overlap, while the class Setosa is linearly separable from the other two.
   (2) *Cancer*: The Wisconsin Breast Cancer data set consists of 683 sample points and each pattern has nine features. There are two categories in the data: malignant and benign. The two classes are known to be linearly separable.
   (3) *Newthyroid*: The original database from where it has been collected is titled as thyroid gland data ("normal", "hypo"and "hyper" functioning). There are three categories in the data: euthyroidism, hypothyroidism and hyperthyroidism. There are a total of 215 instances and the number of attributes is five.
   (4) *Wine*: This data set has 178 points along with 13 features resulting from a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. It is divided into three clusters.
   (5) *LiverDisorder*: The Liver Disorder data set consists of 345 instances having 6 features each. The data set has two categories.
   (6) *LungCancer*: This data set consists of 32 instances having 56 features each. The data set describes 3 types of pathological lung cancers.
   (7) *Glass*: This data set has 214 points having 9 features (id number, Refractive index, Sodium, Magnesium, Aluminum, Silicon, Potassium, Calcium, Barium and Iron). There are 6 categories present in this data set.
   (8) *Yeast*: The Yeast data set consists of 1484 instances having 8 features each. The data set has ten categories.
   (9) *Diabetes*: This data set has 768 points having 8 features. There are 2 categories present in this data set.
   (10) *CMC*: This data set consists of 1473 sample points and each pattern has nine features. There are three categories in the data set.

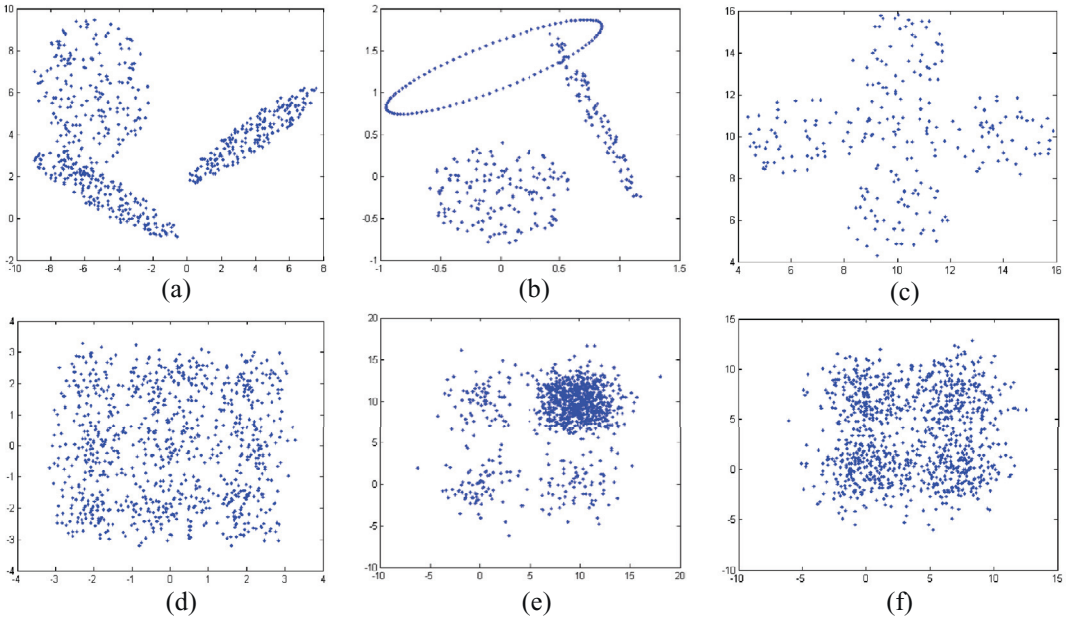

Fig. 3. Artificial data sets: (a) Sym_3_2, (b) Ring_3_2, (c) Sph_5_2, (d) Sph_9_2, (e) Sizes_5, and (f) Square4.

In the experiments, *MOFC-TMS* was implemented based on the following parameters: q=5, m=20, t_{max}=100, C_r=0.8. In this section, *MOFC-TMS* is firstly compared with three multi-objective clustering approaches, and then is compared with several single-objective clustering approaches, including its single-objective versions. To evaluate the performance of these clustering algorithms quantitatively, two validation measures, *Minkowski score* and *F-measure*, were computed in experiments. Let m_{ij} be the number of points which belong to both cluster i and cluster j, m_i the total number of points in cluster i. The *F-measure* of cluster i with respect to class j is defined by

$$F(i,j) = \frac{(2 \times precision(i,j) \times recall(i,j))}{(precision(i,j) + recall(i,j))} \tag{9}$$

where $precision(i,j) = p_{ij} = m_{ij}/m_i$ expresses the precision of cluster i with respect to class j, and $recall(i,j) = m_{ij}/m_j$ denotes the recall of cluster i with respect to class j. Thus, the overall *F-measure* of the whole partitioning is calculated by

$$F = \sum_j \frac{m_j}{m} \max_i F(i,j) \tag{10}$$

For *F-measure*, the optimum score is 1, with higher scores being better.

5.1. Comparison with multi-objective clustering approaches

Three multi-objective clustering approaches were chosen to compare with *MOFC-TMS*, including *VAMOSA* [26], *GenClustMOO* [27] and *MOmoDEFC* [28]. The three multi-objective clustering approaches have been realized in experiments based on the parameters used in literature. These parameters are provided as follows. For *GenClustMOO*, $CL = 100$, $HL = 50$, $iter = 50$, $T_{max} = 100$, $T_{min} = 0.00001$ and $\alpha = 0.9$. Parameters of *MOmoDEFC* are given by population size $= 50$, the number of iterations $= 100$, $\alpha = 0.8$ and $F = 1/(1 + \exp(-(1/\text{generation})))$. For *VAMOSA*, $T_{max} = 100$, $T_{min} = 0.00001$, $\alpha = 0.8$, $HL = 100$ and $SL = 200$.

Table 1 reports the comparison results of four multi-objective clustering approaches on six artificial and ten real-life data sets in terms of *Minkowski score* and *F-measure*. The comparison results are the average values and standard deviations of these algorithms over 30 runs. The performance comparison of these multi-objective clustering approaches is provided as follows.

(1) *MOFC-TMS* vs. *GenClustMOO*.
For *Sym_3_2* and *Ring_3_2* that contain symmetrical shaped clusters, *MOFC-TMS* and *GenClustMOO* have similar performance. For three highly overlapping data sets, *Sph_5_2, Sph_9_2* and *Iris, MOFC-TMS* obviously outperforms *GenClustMOO*. However, *GenClustMOO* is slightly better than *MOFC-TMS* on *Sizes_5, Square4, Cancer, Wine* and *Glass*. For *Newthyroid, LiverDisorder* and *LungCancer, MOFC-TMS* is slightly better than *GenClustMOO*. For *Yeast* and *Diabetes, MOFC-TMS* is obviously better than *GenClustMOO*, while for *CMC* they have similar performance.

(2) *MOFC-TMS* vs. *MOmoDEFC*.
For *Sym_3_2* and *Ring_3_2, MOFC-TMS* and *MOmoDEFC* have similar performance. Moreover, the performance of *MOFC-TMS* is better than that of *MOmoDEFC* on other data sets.

(3) *MOFC-TMS* vs. *VAMOSA*.
For all of thirteen data sets, the performance of *MOFC-TMS* is obviously superior to that of *VAMOSA*.

5.2. Comparison with single-objective clustering approaches

In the experiments, *MOFC-TMS* was compared with two single-objective clustering approaches (*DE*-based and *PSO*-based clustering approaches) and *FCM*. The parameters of *DE* are: population size $= 50$, the number of iterations $= 100$, $\alpha = 0.8$ and $F = 1/(1 + \exp(-(1/\text{generation})))$. For *PSO*, its parameters are chosen as follows: population size $= 50$, the number of iterations $= 100$, $c_1 = c_2 = c_3 = 0$, $w_{max} = 0.9$ and $w_{min} = 0.2$.

Three single-objective versions of *MOFC-TMS* have been implemented in the experiments, which correspond to optimizing one of J_m, *XB* and *Sym* respectively, denoted by *TMS(J_m), TMS(XB)* and *TMS(Sym)*. These single-objective versions are similar to *Fuzzy-MC* [50], but they use the same tissue-like membrane system with a special structure and differential evolution mechanism as *MOFC-TMS*. In some senses, these single-objective versions can be viewed as an improvement of *Fuzzy-MC*. In the realization, *TMS(J_m), TMS(XB)* and *TMS(Sym)* used the same parameters with *MOFC-TMS*.

Table 1
Comparison results of the proposed clustering approach with three multi-objective clustering approaches in terms of Minkowski score and F-measure.

Data sets	*MOFC-TMS*		*GenClustMOO*		*MOmoDEFC*		*VAMOSA*	
	MS	*FM*	*MS*	*FM*	*MS*	*FM*	*MS*	*FM*
Sym_3_2	**0.193** ± **0.011**	0.958 ± **0.009**	0.208 ± 0.025	**0.963** ± 0.023	0.215 ± 0.029	0.942 ± 0.027	0.268 ± 0.027	0.884 ± 0.035
Ring_3_2	**0.093** ± **0.012**	0.952 ± **0.013**	0.098 ± 0.019	**0.959** ± 0.021	0.104 ± 0.028	0.946 ± 0.023	0.256 ± 0.036	0.879 ± 0.037
Sph_5_2	**0.056** ± **0.012**	**0.961** ± **0.011**	0.228 ± 0.024	0.943 ± 0.025	0.098 ± 0.033	0.954 ± 0.021	0.215 ± 0.032	0.783 ± 0.033
Sph_9_2	**0.284** ± **0.013**	**0.752** ± **0.012**	0.491 ± 0.023	0.673 ± 0.019	0.352 ± 0.032	0.705 ± 0.022	0.638 ± 0.038	0.558 ± 0.032
Sizes_5	0.176 ± **0.008**	0.959 ± **0.014**	**0.174** ± 0.021	**0.961** ± 0.029	0.237 ± 0.031	0.932 ± 0.024	0.187 ± 0.031	0.942 ± 0.029
Square4	0.251 ± **0.011**	0.914 ± **0.012**	**0.249** ± 0.021	**0.916** ± 0.022	0.279 ± 0.032	0.873 ± 0.023	0.541 ± 0.036	0.795 ± 0.034
Iris	**0.219** ± **0.009**	**0.836** ± **0.013**	0.362 ± 0.025	0.823 ± 0.028	0.268 ± 0.035	0.812 ± 0.021	0.815 ± 0.033	0.682 ± 0.032
Cancer	0.085 ± **0.013**	0.970 ± **0.014**	**0.087** ± 0.022	**0.969** ± 0.013	0.095 ± 0.029	0.938 ± 0.023	0.352 ± 0.035	0.816 ± 0.035
Newthyroid	**0.278** ± **0.011**	**0.852** ± **0.012**	0.283 ± 0.023	0.849 ± 0.025	0.294 ± 0.028	0.836 ± 0.022	0.594 ± 0.032	0.781 ± 0.033
Wine	0.345 ± **0.013**	0.696 ± **0.013**	**0.343** ± 0.022	**0.698** ± 0.029	0.351 ± 0.035	0.684 ± 0.025	0.981 ± 0.043	0.516 ± 0.032
LiverDisorder	**0.342** ± **0.011**	**0.681** ± **0.013**	0.348 ± 0.023	0.675 ± 0.025	0.352 ± 0.034	0.662 ± 0.021	0.986 ± 0.039	0.463 ± 0.034
LungCancer	**0.326** ± **0.010**	**0.819** ± **0.012**	0.332 ± 0.022	0.815 ± 0.028	0.339 ± 0.031	0.809 ± 0.022	0.862 ± 0.037	0.692 ± 0.033
Glass	0.260 ± **0.012**	0.501 ± **0.013**	**0.258** ± 0.021	**0.503** ± 0.029	0.281 ± 0.031	0.468 ± 0.023	0.561 ± 0.033	0.415 ± 0.032
Yeast	**0.361** ± **0.013**	**0.585** ± **0.011**	0.368 ± 0.022	0.582 ± 0.027	0.372 ± 0.033	0.536 ± 0.022	0.594 ± 0.035	0.451 ± 0.031
Diabetes	**0.285** ± **0.012**	**0.712** ± **0.013**	0.291 ± 0.021	0.703 ± 0.024	0.315 ± 0.032	0.684 ± 0.023	0.436 ± 0.032	0.322 ± 0.032
CMC	0.425 ± **0.012**	**0.653** ± **0.014**	**0.421** ± 0.023	0.647 ± 0.028	0.438 ± 0.033	0.632 ± 0.021	0.514 ± 0.035	0.593 ± 0.032

Table 2
Comparison results of the proposed clustering approach with three single-objective clustering approaches and FCM in terms of Minkowski score and F-measure

Data sets	Indices	*MOFC-TMS*	*TMS(J_m)*	*TMS(XB)*	*TMS(Sym)*	*DE*	*PSO*	*FCM*
Sym_3_2	*MS*	**0.193** ± **0.011**	0.231 ± 0.022	0.264 ± 0.023	0.259 ± 0.019	0.325 ± 0.022	0.342 ± 0.032	0.485 ± 0.043
	FM	**0.958** ± **0.009**	0.936 ± 0.018	0.946 ± 0.019	0.932 ± 0.022	0.914 ± 0.023	0.865 ± 0.022	0.732 ± 0.032
Ring_3_2	*MS*	**0.093** ± **0.012**	0.225 ± 0.023	0.237 ± 0.022	0.254 ± 0.023	0.328 ± 0.021	0.364 ± 0.033	0.406 ± 0.042
	FM	**0.952** ± **0.013**	0.917 ± 0.020	0.932 ± 0.021	0.905 ± 0.018	0.847 ± 0.022	0.829 ± 0.024	0.789 ± 0.033
Sph_5_2	*MS*	**0.056** ± **0.012**	0.149 ± 0.019	0.182 ± 0.023	0.165 ± 0.021	0.285 ± 0.023	0.313 ± 0.031	0.458 ± 0.041
	FS	**0.961** ± **0.011**	0.913 ± 0.021	0.938 ± 0.020	0.926 ± 0.023	0.813 ± 0.019	0.795 ± 0.023	0.669 ± 0.031
Sph_9_2	*MS*	**0.284** ± **0.013**	0.341 ± 0.023	0.338 ± 0.024	0.319 ± 0.022	0.415 ± 0.022	0.452 ± 0.032	0.531 ± 0.042
	FM	**0.752** ± **0.012**	0.735 ± 0.022	0.738 ± 0.022	0.719 ± 0.019	0.645 ± 0.020	0.612 ± 0.022	0.503 ± 0.032
Sizes_5	*MS*	**0.176** ± **0.008**	0.284 ± 0.021	0.261 ± 0.021	0.235 ± 0.021	0.324 ± 0.023	0.351 ± 0.033	0.483 ± 0.041
	FM	**0.959** ± **0.014**	0.879 ± 0.020	0.884 ± 0.023	0.912 ± 0.020	0.856 ± 0.021	0.837 ± 0.024	0.792 ± 0.031
Square4	*MS*	**0.251** ± **0.011**	0.346 ± 0.019	0.338 ± 0.022	0.299 ± 0.023	0.382 ± 0.022	0.439 ± 0.032	0.528 ± 0.042
	FM	**0.914** ± **0.012**	0.829 ± 0.021	0.836 ± 0.022	0.854 ± 0.021	0.738 ± 0.022	0.704 ± 0.021	0.635 ± 0.032
Iris	*MS*	**0.219** ± **0.009**	0.356 ± 0.020	0.325 ± 0.021	0.371 ± 0.022	0.427 ± 0.021	0.468 ± 0.031	0.536 ± 0.039
	FM	**0.836** ± **0.013**	0.803 ± 0.022	0.814 ± 0.018	0.795 ± 0.022	0.774 ± 0.021	0.758 ± 0.023	0.715 ± 0.033
Cancer	*MS*	**0.085** ± **0.013**	0.127 ± 0.022	0.154 ± 0.022	0.216 ± 0.023	0.259 ± 0.022	0.281 ± 0.031	0.385 ± 0.043
	FM	**0.970** ± **0.014**	0.905 ± 0.023	0.921 ± 0.023	0.894 ± 0.023	0.856 ± 0.023	0.829 ± 0.022	0.738 ± 0.034
Newthyroid	*MS*	**0.278** ± **0.011**	0.356 ± 0.021	0.331 ± 0.022	0.368 ± 0.022	0.427 ± 0.023	0.445 ± 0.032	0.538 ± 0.042
	FM	**0.852** ± **0.012**	0.833 ± 0.018	0.831 ± 0.021	0.816 ± 0.019	0.758 ± 0.021	0.734 ± 0.023	0.661 ± 0.032
Wine	*MS*	**0.345** ± **0.013**	0.439 ± 0.023	0.412 ± 0.025	0.428 ± 0.024	0.573 ± 0.021	0.607 ± 0.034	0.751 ± 0.045
	FM	**0.696** ± **0.013**	0.618 ± 0.023	0.637 ± 0.023	0.609 ± 0.023	0.654 ± 0.022	0.517 ± 0.021	0.445 ± 0.033
LiverDisorder	*MS*	**0.342** ± **0.011**	0.435 ± 0.022	0.417 ± 0.021	0.426 ± 0.021	0.535 ± 0.023	0.572 ± 0.033	0.638 ± 0.041
	FM	**0.681** ± **0.013**	0.634 ± 0.019	0.649 ± 0.020	0.625 ± 0.020	0.612 ± 0.019	0.598 ± 0.023	0.563 ± 0.032
LungCancer	*MS*	**0.326** ± **0.010**	0.428 ± 0.021	0.451 ± 0.023	0.436 ± 0.023	0.485 ± 0.021	0.513 ± 0.032	0.589 ± 0.042
	FM	**0.819** ± **0.012**	0.762 ± 0.022	0.784 ± 0.022	0.772 ± 0.021	0.663 ± 0.020	0.635 ± 0.021	0.584 ± 0.031
Glass	*MS*	**0.260** ± **0.012**	0.369 ± 0.023	0.347 ± 0.024	0.347 ± 0.022	0.482 ± 0.023	0.513 ± 0.033	0.614 ± 0.043
	FM	**0.501** ± **0.013**	0.482 ± 0.024	0.486 ± 0.023	0.479 ± 0.023	0.463 ± 0.022	0.448 ± 0.024	0.415 ± 0.033
Yeast	*MS*	0.361 ± **0.013**	0.368 ± 0.022	**0.357** ± 0.024	0.365 ± 0.022	0.371 ± 0.021	0.415 ± 0.023	0.492 ± 0.032
	FM	**0.585** ± **0.011**	0.581 ± 0.023	0.578 ± 0.022	0.583 ± 0.021	0.562 ± 0.023	0.481 ± 0.025	0.435 ± 0.035
Diabetes	*MS*	0.285 ± **0.012**	0.289 ± 0.021	**0.282** ± 0.022	0.285 ± 0.023	0.292 ± 0.024	0.313 ± 0.021	0.386 ± 0.033
	FM	0.792 ± **0.013**	0.791 ± 0.022	**0.795** ± 0.021	0.789 ± 0.024	0.765 ± 0.022	0.723 ± 0.031	0.691 ± 0.041
CMC	*MS*	**0.425** ± **0.012**	0.471 ± 0.023	0.459 ± 0.022	0.448 ± 0.0243	0.483 ± 0.022	0.536 ± 0.033	0.581 ± 0.042
	FM	**0.653** ± **0.014**	0.632 ± 0.025	0.647 ± 0.023	0.641 ± 0.023	0.629 ± 0.024	0.591 ± 0.032	0.563 ± 0.043

Table 2 reports the comparison results of *MOFC-TMS* with *TMS, DE, PSO* and *FCM* on six artificial and ten real-life data sets in terms of *Minkowski score* and *F-measure*. The presented results are the average values and standard deviations of the approaches over 30 runs. The comparison of *MOFC-TMS* with *TMS, DE, PSO* and *FCM* is analyzed as follows.

(1) *MOFC-TMS* vs. *TMS*.
For *Sym_3_2* and *Ring_3_2, MOFC-TMS* are superior to *TMS(J_m), TMS(XB)* and *TMS(Sym)* in terms of both *MS* and *FM* indices, and *TMS(XB)* and *TMS(Sym)* are better than *TMS(J_m)*. The comparison results illustrate that *MOFC-TMS* can well deal with the data sets with symmetrical shaped clusters. For three highly overlapping data sets, *Sph_5_2, Sph_9_2* and *Iris, MOFC-TMS* obviously outperforms *TMS(J_m), TMS(XB)* and *TMS(Sym)*. Therefore, *MOFC-TMS* can achieve a satisfactory clustering performance on highly overlapping data sets. For *Sizes_5* and *Square4*, results of *MOFC-TMS* are better than that of *TMS(J_m), TMS(XB)* and *TMS(Sym)*, which illustrate that *MOFC-TMS* has a good performance to cluster the data sets with well-separated clusters. However, for *Yeast* and *Diabetes, TMS(XB)* is slightly better than *MOFC-TMS*. Furthermore, the comparison results on other data sets clearly show that *MOFC-TMS* can also achieve a good partitioning in comparison to *TMS(J_m), TMS(XB)* and *TMS(Sym)*.

(2) *MOFC-TMS* vs. *DE, PSO* and *FCM*.
Looking at the results of *MS* index, it can be clearly seen that for each data set, *MOFC-TMS* has the lowest mean value, second is *DE*, third is *PSO*, and the largest is *FCM*. Moreover, *MOFC-TMS*

Table 3
The p-values produced by Wilcoxon's rank sum test comparing with other approaches in terms of Minkowski score.

Data sets	*MOFC-TMS vs.*								
	GenClustMOO	*MOmoDEFC*	*VAMOSA*	*DE*	*PSO*	*FCM*	*TMS(J_m)*	*TMS(XB)*	*TMS(Sym)*
Sym_3_2	0.0419 (+)	0.0338 (+)	0.0189 (+)	6.52e−3 (+)	5.23e−3 (+)	1.04e−4 (+)	0.0342 (+)	0.0346 (+)	0.0345 (+)
Ring_3_2	0.0449 (+)	0.0439 (+)	4.23e−3 (+)	7.61e−4 (+)	2.56e−4 (+)	1.02e−4 (+)	0.0441 (+)	0.0443 (+)	0.0444 (+)
Sph_5_2	2.86e−3 (+)	0.0347 (+)	6.84e−3 (+)	6.13e−4 (+)	4.95e−4 (+)	1.51e−5 (+)	0.0362 (+)	0.0371 (+)	0.0365 (+)
Sph_9_2	1.03e−3 (+)	0.0142 (+)	6.61e−5 (+)	7.19e−3 (+)	3.53e−3 (+)	5.46e−4 (+)	0.0141 (+)	0.0142 (+)	0.0135 (+)
Sizes_5	0.0519 (−)	0.0228 (+)	0.0407 (+)	7.62e−3 (+)	2.02e−3 (+)	8.27e−5 (+)	0.0245 (+)	0.0242 (+)	0.0228 (+)
Square4	0.0522 (−)	0.0419 (+)	1.07e−4 (+)	9.04e−3 (+)	4.72e−3 (+)	2.64e−4 (+)	0.0432 (+)	0.0431 (+)	0.0420 (+)
Iris	4.92e−3 (+)	0.0309 (+)	1.32e−7 (+)	8.98e−4 (+)	4.74e−4 (+)	4.82e−5 (+)	0.0341 (+)	0.0327 (+)	0.0331 (+)
Cancer	0.0468 (+)	0.0412 (+)	3.25e−4 (+)	3.75e−3 (+)	1.19e−3 (+)	3.26e−4 (+)	0.0431 (+)	0.0437 (+)	0.0441 (+)
Newthyroid	0.0453 (+)	0.0463 (+)	7.98e−5 (+)	6.14e−3 (+)	3.35e−3 (+)	4.51e−5 (+)	0.0476 (+)	0.0471 (+)	0.0477 (+)
Wine	0.0524 (−)	0.0452 (+)	1.13e−7 (+)	6.81e−4 (+)	3.41e−4 (+)	8.22e−5 (+)	0.0465 (+)	0.0458 (+)	0.0463 (+)
LiverDisorder	0.0462 (+)	0.0375 (+)	1.06e−7 (+)	1.03e−3 (+)	6.49e−4 (+)	5.45e−4 (+)	0.0419 (+)	0.0408 (+)	0.0414 (+)
LungCancer	0.0428 (+)	0.0399 (+)	6.29e−7 (+)	5.96e−3 (+)	1.05e−3 (+)	2.63e−4 (+)	0.0423 (+)	0.0431 (+)	0.0428 (+)
Glass	0.0521 (−)	0.0296 (+)	1.39e−4 (+)	7.38e−4 (+)	4.05e−4 (+)	4.29e−5 (+)	0.0385 (+)	0.0369 (+)	0.0423 (+)
Yeast	0.0439 (+)	0.0382 (+)	4.82e−7 (+)	3.56e−3 (+)	4.15e−4 (+)	3.29e−4 (+)	0.0410 (+)	0.0508 (−)	0.0452 (+)
Diabetes	0.0415 (+)	0.0275 (+)	2.98e−7 (+)	4.26e−3 (+)	4.56e−4 (+)	2.18e−4 (+)	0.0491 (+)	0.0501 (−)	0.0493 (+)
CMC	0.0534 (−)	0.0448 (+)	1.26e−6 (+)	7.29e−4 (+)	3.82e−4 (+)	5.72e−5 (+)	0.0458 (+)	0.0455 (+)	0.0452 (+)

Table 4
The p-values produced by Wilcoxon's rank sum test comparing with other approaches in terms of F-measure.

Data sets	*MOFC-TMS vs.*								
	GenClustMOO	*MOmoDEFC*	*VAMOSA*	*DE*	*PSO*	*FCM*	*TMS(J_m)*	*TMS(XB)*	*TMS(Sym)*
Sym_3_2	0.0416 (+)	3.22e−3 (+)	2.51e−3 (+)	0.0318 (+)	0.0108 (+)	7.95e−4 (+)	3.21e−3 (+)	3.27e−3 (+)	3.19e−3 (+)
Ring_3_2	0.0394 (+)	4.41e−3 (+)	1.92e−3 (+)	8.94e−3 (+)	8.16e−3 (+)	2.83e−3 (+)	4.27e−3 (+)	4.38e−3 (+)	4.23e−3 (+)
Sph_5_2	0.0328 (+)	4.06e−3 (+)	2.19e−3 (+)	4.69e−3 (+)	5.37e−3 (+)	3.72e−5 (+)	3.94e−3 (+)	4.01e−3 (+)	3.98e−3 (+)
Sph_9_2	0.0194 (+)	2.35e−3 (+)	8.78e−4 (+)	8.82e−3 (+)	6.24e−3 (+)	4.86e−4 (+)	3.31e−3 (+)	3.39e−3 (+)	2.85e−3 (+)
Sizes_5	0.0518 (−)	3.81e−3 (+)	3.95e−3 (+)	0.0138 (+)	7.52e−3 (+)	3.97e−3 (+)	3.61e−3 (+)	3.65e−3 (+)	3.76e−3 (+)
Square4	0.0523 (−)	3.17e−3 (+)	7.64e−3 (+)	3.95e−3 (+)	1.12e−3 (+)	2.93e−4 (+)	3.04e−3 (+)	3.07e−3 (+)	3.12e−3 (+)
Iris	0.0357 (+)	2.92e−3 (+)	3.72e−3 (+)	0.0319 (+)	0.0193 (+)	8.45e−3 (+)	2.89e−3 (+)	2.93e−3 (+)	2.87e−3 (+)
Cancer	0.0451 (+)	3.94e−3 (+)	2.96e−3 (+)	0.0154 (+)	8.62e−3 (+)	9.87e−4 (+)	3.83e−3 (+)	3.85e−3 (+)	3.81e−3 (+)
Newthyroid	0.0418 (+)	4.16e−3 (+)	2.38e−3 (+)	9.68e−3 (+)	0.0417 (+)	1.12e−3 (+)	4.13e−3 (+)	4.11e−3 (+)	4.06e−3 (+)
Wine	0.0523 (−)	3.71e−3 (+)	2.15e−3 (+)	0.0394 (+)	5.76e−3 (+)	4.57e−4 (+)	3.65e−3 (+)	3.68e−3 (+)	3.63e−3 (+)
LiverDisorder	0.0382 (+)	3.18e−3 (+)	4.81e−4 (+)	0.0284 (+)	0.0118 (+)	8.86e−3 (+)	3.12e−3 (+)	3.14e−3 (+)	3.11e−3 (+)
LungCancer	0.0483 (+)	4.65e−3 (+)	8.53e−3 (+)	4.86e−3 (+)	3.27e−3 (+)	6.94e−4 (+)	4.55e−3 (+)	4.61e−3 (+)	4.59e−3 (+)
Glass	0.0519 (−)	3.41e−3 (+)	1.72e−3 (+)	0.0294 (+)	0.0315 (+)	0.0281 (+)	3.54e−3 (+)	3.55e−3 (+)	3.52e−3 (+)
Yeast	0.0392 (+)	4.27e−3 (+)	5.38e−4 (+)	3.95e−3 (+)	2.83e−3 (+)	7.43e−4 (+)	4.39e−3 (+)	4.35e−3 (+)	4.31e−3 (+)
Diabetes	0.0428 (+)	4.19e−3 (+)	3.85e−4 (+)	2.87e−3 (+)	2.36e−3 (+)	4.91e−4 (+)	0.492 (+)	0.502 (−)	0.489 (+)
CMC	0.0473 (+)	4.38e−3 (+)	2.76e−3 (+)	0.0379 (+)	0.0235 (+)	3.59e−3 (+)	4.39e−3 (+)	4.45e−3 (+)	4.42e−3 (+)

obtains the highest mean value of *FM* on each data set, whereas *FCM* is the lowest in the four approaches. These comparison results show that *MOFC-TMS* has a good clustering performance over *DE, PSO* and *FCM*.

5.3. Stability comparison

Table 1 provides standard deviations of the proposed and compared multi-objective clustering approaches over 30 runs in terms of *MS* and *FM*, respectively. For all of data sets, standard deviations of *MOFC-TMS* are lower than those of *GenClustMOO, MOmoDEFC* and *VAMOSA*. It can be also seen that for *Sizes_5, Square4, Cancer, Wine* and *Glass*, standard deviations of *MOFC-TMS* are still lower than those of *GenClustMOO* although average values of *GenClustMOO* is slightly better than those of *MOFC-TMS*. Table 2 compares standard deviations of *MOFC-TMS* with several single-objective clustering approaches. In contrast to these single-objective clustering approaches, *MOFC-TMS* has the smallest standard deviation on each data set. The comparison results demonstrate that *MOFC-TMS* outperforms the compared multi-objective and single-objective clustering approaches in terms of stability.

5.4. Statistical significance test

The Wilcoxon's rank sum test, a nonparametric statistical significance test for independent samples, has been conducted at the 5% significance level in the experiments. We created seven groups for each data set, which correspond to the ten approaches (*MOFC-TMS, GenClustMOO, MOmoDEFC, VAMOSA, DE, PSO, FCM*), *TMS(J_m), TMS(XB)* and *TMS(Sym)*, respectively. Each group consists of *MS* and *FM* values produced by 30 consecutive runs of the corresponding approaches on these data sets, respectively. The mean values of *MS* and *FM* for the seven groups are provided in Tables 1 and 2, respectively. The shown results indicate that most of mean values of *MOFC-TMS* are better than those of the other approaches. To prove the goodness is statistically significant, a statistical significance test has been completed on the six artificial data sets and ten real-life data sets. Tables 3 and 4 provide the p-values of two groups (one group corresponding to *MOFC-TMS* and another group corresponding to some other approach) in terms of *MS* and *FM*, respectively. Note that the symbols "+" and "-" represent significant difference and no significant difference, respectively. It is evident from Tables 3 and 4 that most of p-values are less than 0.05 (5% significance level) except the five cases of *MOFC-TMS* vs. *GenClustMOO* on *Sizes_5, Square4, Wine, Glass* and *CMC* and the two cases of *MOFC-TMS* vs. *TMS(XB)* on *Yeast* and *Diabetes*. This is a strong evidence against the null hypothesis, establishing significant superiority of the proposed *MOFC-TMS*.

6. Conclusions

This paper discussed the use of tissue-like membrane systems to develop a multiobjective clustering framework for fuzzy clustering problem. A tissue-like membrane system with a special membrane structure has been designed to integrate the differential

evolution mechanism and the non-dominated sorting technique. By optimizing three widely used cluster validity indices simultaneously, the multiobjective clustering framework realized a novel fuzzy clustering approach. The proposed approach has been evaluated on six artificial and ten real-life data sets, and has been compared with three recently developed multiobjective clustering approaches and three single-objective clustering approaches as well as FCM. Comparison results clearly exhibit the advantage of the proposed multiobjective clustering approach on solving fuzzy clustering problem.

In recent years, several works have proved that evolutionary and other types of optimization techniques can be used to find the number of clusters automatically for fuzzy partitions [24,25,57]. The proposed multiobjective clustering framework cannot deal with this situation because it uses a tissue-like membrane system with a fixed membrane structure. However, some membrane systems with dynamic membrane structure seem to be suitable to deal with this situation, for example, membrane systems with cell division. So, another further work is using membrane systems with dynamic membrane structures to optimize several objective functions and at the same time find the optimal number of clusters.

Acknowledgment

This work was partially supported by the National Natural Science Foundation of China (Grant No. 61472328), Chunhui Project Found of the Education Department of China (Nos. Z2016143 and Z2016148), and Research Found of the Education Department of Sichuan province (No. 17TD0034), China.

References

[1] B. Everitt, S. Landau, M. Leese, Cluster analysis, Arnold, London, 2001.
[2] G. Gan, C. Ma, J. Wu, Data clustering: Theory, 2007. Algorithms, and Applications, SIAM.
[3] J.C. Bezdek, Pattern recognition with fuzzy objective function algorithm, Plenum, New York, 1981.
[4] A.K. Jain, Data clustering: 50 years beyond k-means, Pattern Recognit. Lett. 31 (2010) 651–666.
[5] S. Das, S. Sil, Kernel-induced fuzzy clustering of image pixels with an improved different evolution algorithm, Inf. Sci. 180 (2010) 1237–1256.
[6] S. Bandyopadhyay, Simulated annealing using a reversible jump Markov chain Monte Carlo algorithm for fuzzy clustering, IEEE Trans. Knowl. Data Eng. 17 (4) (2005) 479–490.
[7] S. Das, A. Konar, Automatic image pixel clustering with an improved differential evolution, Appl. Soft Comput. 9 (2009) 226–236.
[8] U. Maulik, I. Saha, Modified differential evolution based fuzzy clustering for pixel classification in remote sensing imagery, Pattern Recognit. 42 (2009) 2135–2149.
[9] U. Maulik, I. Saha, Automatic fuzzy clustering using modified differential evolution for image classification, IEEE Trans. Geosci. Remote Sens. 48 (9) (2010) 3503–3510.
[10] A. Hatamlou, Black hole: a new heuristic optimization approach for data clustering, Inf. Sci. (222) (2013) 175–184.
[11] A. Hatamlou, M. Hatamlou, PSOHS: an efficient two-stage approach for data clustering, Memetic Comput. 5 (2) (2013) 155–161.
[12] Y.T. Kao, E. Zahara, I.W. Kao, A hybridized approach to data clustering, Expert Syst. Appl. 34 (3) (2008) 1754–1762.
[13] R.J. Kuo, Y.J. Syu, Z.-Y. Chen, F.C. Tien, Integration of particle swarm optimization and genetic algorithm for dynamic clustering, Inf. Sci. 195 (2012) 124–140.
[14] D. Karaboga, C. Ozturk, A novel clustering approach: artificial bee colony (ABC) algorithm, Appl. Soft Comput. 11 (2011) 652–657.
[15] U. Maulik, S. Bandyopadhyay, Genetic algorithm based clustering technique, Pattern Recognit. 33 (2000) 1455–1465.
[16] U. Maulik, S. Bandyopadhyay, Fuzzy partitioning using a real-coded variable-length genetic algorithm for pixel classification, IEEE Trans. Geosci. Remote Sens. 41 (5) (2003) 1075–1081.
[17] C.D. Nguyen, K.J. Cios, GAKREM: a novel hybrid clustering algorithm, Inf. Sci. 178 (2008) 4205–4227.
[18] P.S. Shelokar, V.K. Jayaraman, B.D. Kulkarni, An ant colony approach for clustering, Anal. Chim. Acta 509 (2) (2004) 187–195.
[19] M.A. Sanchez, O. Castillo, J.R. Castro, P. Melin, Fuzzy granular gravitational clustering algorithm for multivariate data, Inf. Sci. 279 (2014) 498–511.
[20] S. Bandyopadhyay, S. Saha, A point symmetry based clustering technique for automatic evolution of clusters, IEEE Trans. Knowl. Data Eng. 20 (11) (2008) 1–17.
[21] U. Maulik, S. Bandyopadhyay, Performance evaluation of some clustering algorithms and validity indices, IEEE Trans. Pattern Anal. Mach. Intell. 24 (12) (2002) 1650–1654.
[22] X.L. Xie, G. Beni, A validity measure for fuzzy clustering, IEEE Trans. Pattern Anal. Mach. Intell. 13 (1991) 841–847.
[23] S. Saha, S. Bandyopadhyay, A new point symmetry based fuzzy genetic clustering technique, Inf. Sci. 179 (2009) 3230–3246.
[24] J. Handl, J. Knowles, An evolutionary approach to multiobjective clustering, IEEE Trans. Evol. Comput. 11 (1) (2007) 56–76.
[25] K. Faceli, M.C.P.D. Souto, D.S.A. Araújo, A.C.P.L.F. Carvalho, Multi-objective clustering ensemble for gene expression data analysis, Neurocomput. 72 (13–15) (2009) 2763–2774.
[26] S. Saha, S. Bandyopadhyay, A symmetry based multiobjective clustering technique for automatic evolution of clusters, Pattern Recognit. 43 (2010) 738–751.
[27] S. Saha, S. Bandyopadhyay, A generalized automatic clustering algorithm in a multiobjective framework, Appl. Soft Comput. 13 (2013) 89–108.
[28] I. Saha, U. Maulik, D. Plewczynski, A new multi-objective technique for differential fuzzy clustering, Appl. Soft Comput. 11 (2011) 2765–2776.
[29] Gh. Păun, Computing with membranes, J. Comput. Syst. Sci. 61 (1) (2000) 108–143.
[30] L. Pan, X. Zeng, X. Zhang, Y. Jiang, Spiking neural P systems with weighted synapses, Neural Process. Lett. 35 (1) (2012) 13–27.
[31] G. Păun, M.J. Pérez-Jiménez, Membrane computing: brief introduction, recent results and applications, BioSyst. 85 (2006) 11–22.
[32] G. Păun, G. Rozenberg, A. Salomaa, The Oxford Handbook of Membrance Computing, Oxford University Press, New York, 2010.
[33] T. Song, L. Pan, G. Păun, Asynchronous spiking neural P systems with local synchronization, Inf. Sci. 219 (2013) 197–207.
[34] X. Zeng, L. Xu, X. Liu, L. Pan, On languages generated by spiking neural P systems with weights, Inf. Sci. 278 (2014) 423–433.
[35] X. Zhang, Y. Liu, B. Luo, L. Pan, Computational power of tissue P systems for generating control languages, Inf. Sci. 278 (10) (2014) 285–297.
[36] B. Song, L. Pan, M.J. Pérez-Jiménez, Cell-like P systems with channel states and symport/antiport rules, IEEE Trans. Nanobiosci. 15 (6) (2016) 555–566.
[37] B. Song, C. Zhang, L. Pan, Tissue-like P systems with evolutional symport/antiport rules, Inf. Sci. 378 (2017) 177–193.
[38] T.Y. Nishida, Membrane algorithms: approximate algorithms for NP-complete optimization problems, in: Ciobanu, et al. (Eds.), Applications of Membrane Computing, Springer, 2006, pp. 303–314.
[39] L. Pan, X. Zeng, Small universal spiking neural P systems working in exhaustive mode, IEEE Trans. Nanobiosci. 10 (2) (2011) 99–105.
[40] H. Peng, J. Wang, M.J. Pérez-Jiménez, H. Wang, J. Shao, T. Wang, Fuzzy reasoning spiking neural P system for fault diagnosis, Inf. Sci. 235 (2013) 106–116.
[41] J. Wang, H. Peng, Adaptive fuzzy spiking neural P systems for fuzzy inference and learning, Int. J. Comput. Math. 90 (4) (2013) 857–868.
[42] J. Wang, P. Shi, H. Peng, M.J. Pérez-Jiménez, T. Wang, Weighted fuzzy spiking neural p systems, IEEE Trans. Fuzzy Syst. 21 (2) (2013) 209–220.
[43] J. Wang, L. Zhou, H. Peng, G.X. Zhang, An extended spiking neural p system for fuzzy knowledge representation, Int. J. Innovative Comput. 7 (7A) (2011) 3709–3724. Information and Control.
[44] L. Huang, I. Suh, A. Abraham, Dynamic multi-objective optimization based on membrane computing for control of time-varying unstable plants, Inf. Sci. 181 (11) (2011) 2370–2391.
[45] H. Peng, Y. Jiang, J. Wang, M.J. Pérez-Jiménez, Membrane clustering algorithm with hybrid evolutionary mechanisms,, J. Softw. 26 (5) (2015) 1001–1012.
[46] H. Peng, J. Wang, M.J. Pérez-Jiménez, A. Riscos-Núñez, The framework of P systems applied to solve optimal watermarking problem, Signal Process. 101 (2014) 256–265.
[47] H. Peng, J. Wang, M.J. Pérez-Jiménez, P. Shi, A novel image thresholding method based on membrane computing and fuzzy entropy, J. Intell. Fuzzy Syst. 24 (2) (2013) 229–237.
[48] J. Wang, P. Shi, H. Peng, Membrane computing model for IIR filter design, Inf. Sci. 329 (2016) 164–176.
[49] G. Zhang, J. Cheng, M. Gheorghe, Q. Meng, A hybrid approach based on different evolution and tissue membrane systems for solving constrained manufacturing parameter optimization problems, Appl. Soft Comput. 13 (3) (2013) 1528–1542.
[50] H. Peng, J. Wang, M.J. Pérez-Jiménez, A. Riscos-Núñez, An unsuper- vised learning algorithm for membrane computing, Inf. Sci. 304 (2015) 80–91.
[51] R. Freund, Gh. Păun, M.J. Pérez-Jiménez, Tissue-like P systems with channel-states, Theor. Comput. Sci. 330 (2005) 101–116.
[52] K. Price, R. Storn, J. Lampinen, Differential Evolution - A Practical Approach to Global Optimization, Springer, Berlin, 2005.
[53] S. Das, P.N. Suganthan, Differential evolution: a survey of the state-of-the-art, IEEE Trans. Evol. Comput. 15 (2011) 4–31.
[54] R. Storn, K. Price, Differential evolution - a simple and efficient heuristic strategy for global optimization over continuous spaces, J. Global Optim. 11 (1997) 341–359.
[55] K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, A fast and elitist multi-objective genetic algorithm: NSGA-II, IEEE Trans. Evol. Comput. 6 (2) (2002) 182–197.
[56] D.N.A. Asuncion, UCI machine learning repository, 2007, http://www.ics.uci.edu/mlearn/MLRepository.html.
[57] D. Horta, I.C. Andrade, R.J.G.B. Campello, Evolutionary fuzzy clustering of relational data, Theor. Comput. Sci. 412 (42) (2011) 5854–5870.